ISSN No:-2456-2165

Improving Warehouse Efficiency Through Automated Counting of Pallets: YOLOv8-Powered Solutions

Praneeth Yennamaneni[1] (Research Associate)
[1]Innodatatics, Hyderabad, India

Vickram R[2] (Research Associate)
[2]Innodatatics, Hyderabad, India

Samyak Sabannawar[3] (Mentor)
[3]Research and Development, Innodatatics, Hyderabad, India.

SreePriya Naroju[4] (Team Leader)
[4]Research and Development, Innodatatics, Hyderabad, India

Bharani Kumar Depuru[5*] (Director)
[5]Innodatatics, Hyderabad, India

Corresponding Author: Bharani Kumar Depuru[5] OCR ID: 0009-0003-4338-8914

Abstract:- This research paper tackles the challenges associated with manual pallet counting within industrial environments and explores the integration of deep learning techniques to enhance operational efficiency. YOLOv8, recognized as a leading object detection algorithm, serves as the foundational framework for this study.

The initial phase involved the extensive collection of images and videos from diverse warehouse settings to curate a comprehensive dataset, instrumental for training and refining the YOLOv8 model. Dataset annotation was meticulously carried out utilising the Roboflow platform. Subsequently, the YOLOv8 model was trained with the custom dataset, achieving an impressive average precision of (insert percentage). The optimal model weights were meticulously saved, facilitating deployment in real-world scenarios.

To extend the practicality of this research, the model was seamlessly integrated into a user-friendly web application powered by Flask, enhancing the accessibility of this technology. The implementation of the model yielded substantial enhancements in efficiency, substantially mitigating the occurrence of operational errors within warehouse management.

This study underscores the remarkable potential of deep learning algorithms, not only within the realm of warehouse management but also in a broad spectrum of real-world applications. The implications of this research extend beyond the domain of pallet counting, illustrating the transformative impact of advanced technology in streamlining industrial processes and contributing to the broader landscape of automation and optimization.

In conclusion, the successful integration of YOLOv8 in this context underscores the transformative power of deep learning, promising ground-breaking solutions for enhanced efficiency and precision in various operational domains.

Keywords:- *Manual Pallet Counting, YOLOv8, Deep Learning Techniques, Warehouse Management, Object Detection, Operational Efficiency, Roboflow.*

I. INTRODUCTION

The primary goal of this research study and the development of an automatic pallet counting system is to improve the accuracy and efficiency of counting the number of pallets in inventory while also reducing the time required for this process. The approach for this research study is based on the Cross-Industry Standard Process for Machine Learning with Quality Assurance , CRISP-ML(Q) methodology, which is conveniently accessible as an open-source framework on the 360DigiTMG website (ak.1) [Fig.1].

ISSN No:-2456-2165

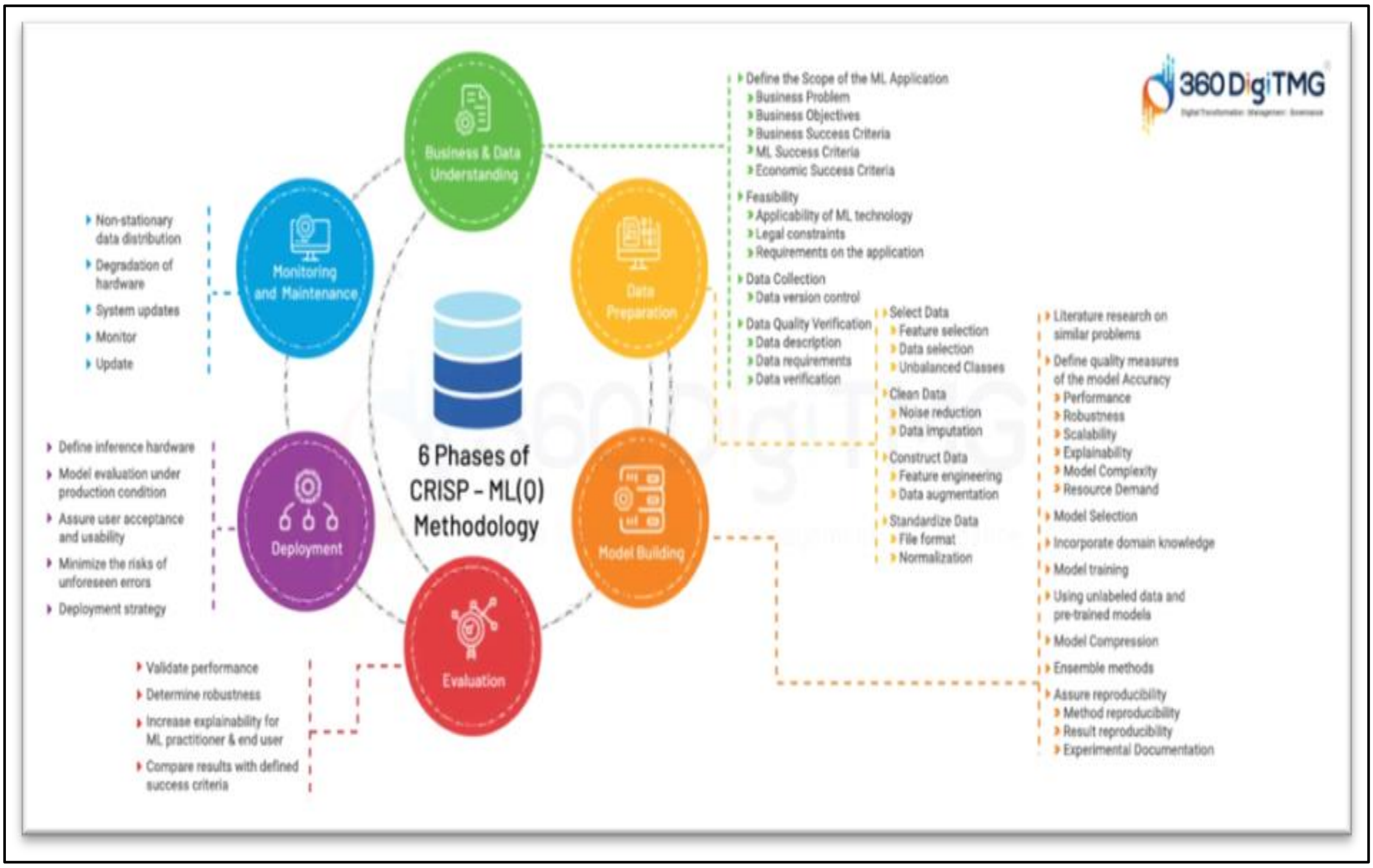

Fig 1 CRISP-ML(Q) - Approach for Quality Assurance for each of the Six Phases. (Source: CRISP-ML(Q) |360DigiTMG)

The realm of modern industrial operations demands precision and efficiency, especially in the logistics and warehousing sector. Within this landscape, the manual counting of pallets poses challenges—consuming valuable human resources and prone to errors that can disrupt the supply chain [1].

This research aims to develop an automatic pallet counting system to enhance accuracy, efficiency, and reduce the time required for this process. Following the CRISP-ML(Q) methodology, we focus on leveraging advanced deep learning techniques, particularly the YOLOv8 object detection algorithm, to revolutionise pallet counting in warehouse environments.

The rationale for this research stems from the significant advancements in deep learning, which have shown remarkable potential in real-world applications [2]. Our approach involves compiling a diverse dataset of warehouse scenarios, meticulously annotated to train and fine-tune the YOLOv8 model for accurate pallet detection and counting.

Anticipated outcomes include substantial improvements in counting efficiency and a reduction in errors associated with manual methods. Furthermore, this study highlights the broader potential of deep learning algorithms in various industrial contexts beyond warehouse management [3].

In subsequent sections, we will delve into methodologies, results, and the implications of our findings, focusing on both pallets counting and the broader landscape of industrial automation and optimization.

This research paper embodies an urgent need for innovative technological solutions in industrial operations. By harnessing the capabilities of the YOLOv8 model, we strive to completely transform pallet counting. YOLOv8's precision, efficiency, and scalability promise a seismic shift in operational dynamics, aiming not only to enhance efficiency but also eliminate human intervention for unparalleled counting accuracy.

Our detailed research encompasses critical components such as comprehensive data collection, model development, and real-world deployment strategies to fully unlock the transformative potential of the YOLOv8 model. By exploring technical details and addressing challenges, we emphasise the synergy between deep learning and pre-processing techniques, ensuring compliance with legal regulations and robustness across various environmental conditions.

This endeavour isn’t just about delivering a scalable solution for pallet counting; it's about unveiling the transformative power of deep learning in optimising industrial processes. The YOLOv8 model serves as the linchpin, driving industries towards a future where efficiency, precision, and cost-effectiveness define operational standards.

II. METHODS AND TECHNIQUES

A. Roboflow

Roboflow is a comprehensive platform designed to streamline the process of preparing image datasets for machine learning and computer vision applications [4]. It offers a range of powerful tools and features that simplify the dataset annotation and management process, making it an invaluable asset for researchers and developers.

Roboflow's user-friendly annotation tools enable efficient labelling of images with bounding boxes, class labels, and other metadata. The platform ensures data consistency, providing standardisation and validation features to maintain dataset quality and reliability [5].

One of Roboflow's standout features is its support for data augmentation, enabling users to enhance dataset diversity and improve the generalisation and robustness of machine learning models. The platform also supports dataset versioning, allowing users to track changes and updates over time.

Roboflow encourages collaboration with multiple team members working on the same dataset simultaneously. Its scalability makes it suitable for handling large datasets efficiently, catering to a wide range of machine learning and computer vision projects.

With its array of features and capabilities, Roboflow has become an essential tool for data preparation in fields such as object detection, image classification, and more. It significantly accelerates the dataset preparation process, contributing to enhanced efficiency and productivity in machine learning and computer vision research and development [6].

B. YOLOv8

YOLOv8, developed by Ultralytics, represents a significant advancement in real-time object detection within the field of computer vision and deep learning. This algorithm is an evolution of the YOLO architecture and is engineered to efficiently and accurately identify and locate objects in images and video streams [7].

Renowned for its state-of-the-art features, YOLOv8 introduces architectural enhancements that substantially enhance both the speed and precision of object detection. Its ability to process entire images in a single pass sets it apart, making it highly effective for real-time applications. YOLOv8 excels in multi-class object detection, enabling the simultaneous identification of diverse object categories within a single frame [8].

What makes YOLOv8 exceptionally versatile is its adaptability and ease of customization. Users can fine-tune the algorithm to meet specific requirements, making it suitable for a wide spectrum of applications, from general object detection to domain-specific tasks.

Open-source in nature, YOLOv8 encourages collaboration and innovation within the computer vision community. It has found utility in various domains, including autonomous vehicles, surveillance systems, and inventory management, offering a powerful solution to complex object detection challenges while contributing to enhanced productivity across industries [9].

C. Flask:

Flask is a lightweight and versatile micro web framework for building web applications in Python [10]. This open-source framework is renowned for its simplicity and flexibility, making it an excellent choice for a wide range of web development projects.

Flask offers a set of essential tools and features for creating web applications while providing developers with the freedom to select and integrate additional libraries and components based on project requirements. It excels in routing, allowing developers to define routes that map to specific functions for handling HTTP requests [11].

With Flask, developers can easily render HTML templates using the Jinja2 template engine, enabling dynamic web page generation by embedding Python code within HTML files. The framework simplifies handling HTTP requests and responses, facilitating the development of web applications [12].

Flask also includes a built-in development server for testing and debugging during the development process. Its ecosystem boasts a wide variety of extensions and libraries that provide additional functionality, such as database integration, user authentication, and more [13].

In summary, Flask's lightweight and minimalistic approach, coupled with its extensibility and simplicity, makes it a popular choice for developing web applications, RESTful APIs, and prototypes. It empowers developers to build web solutions with ease, adaptability, and efficiency, contributing to the success of web development projects across diverse industries.

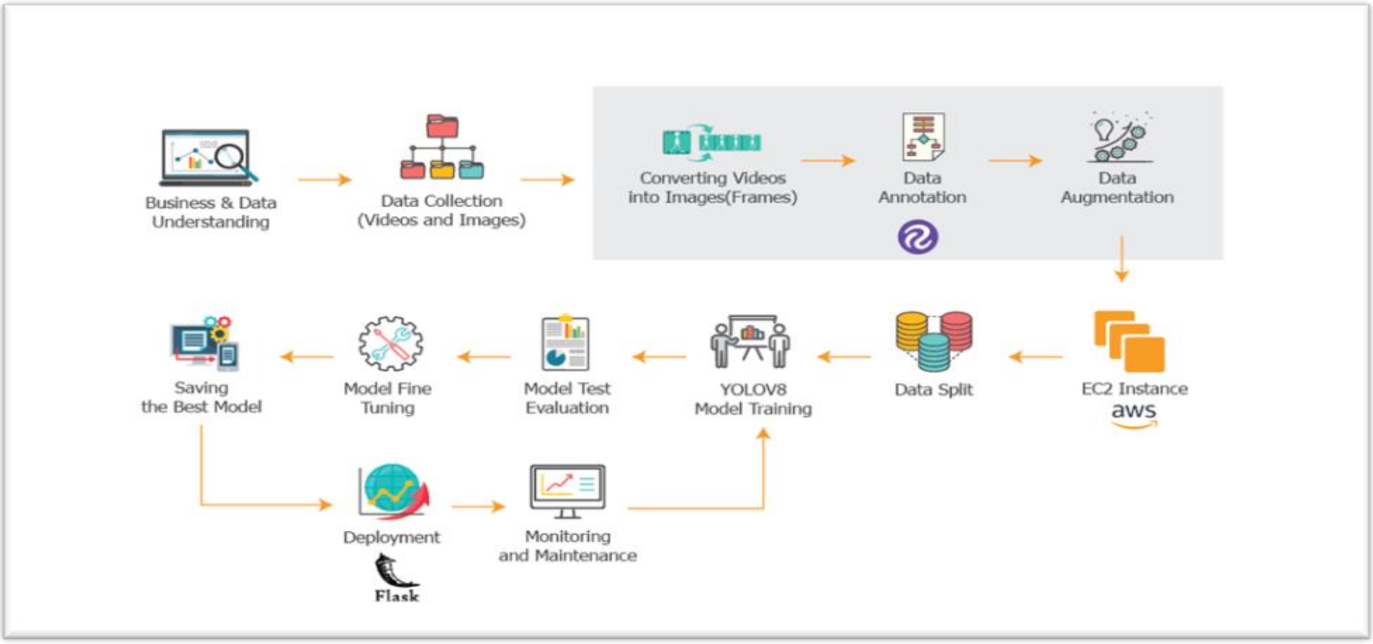

Fig 2 Workflow of Methodology - Framework and Training Process of Object Detection using YOLOv8 Model
(Source: - ML Workflow - 360DigiTMG)

D. Workflow of Methodology:

➢ *Step 1: Data Collection*

Client Data Consists of one video of 26 minutes and 6 videos of 1-2 minutes and 330 images. Video (26 minutes) converted into 4000 images approx. Total 5227 images were taken for POC.

- *Data Formats:*

✓ Image Formats: JPEG
✓ Video Formats: MOV (used to extract frames)

➢ *Step 2: Upload Image or Video*

- Users can upload image or video format
- If a video is uploaded, it is converted into images for further processing.

➢ *Step 3: Annotation and Basic Pre-processing and Augmentation*

- *Annotation*

✓ The images are then uploaded to Roboflow to perform annotation. Bounding Boxes are drawn around the pallets in the images [Fig.3]

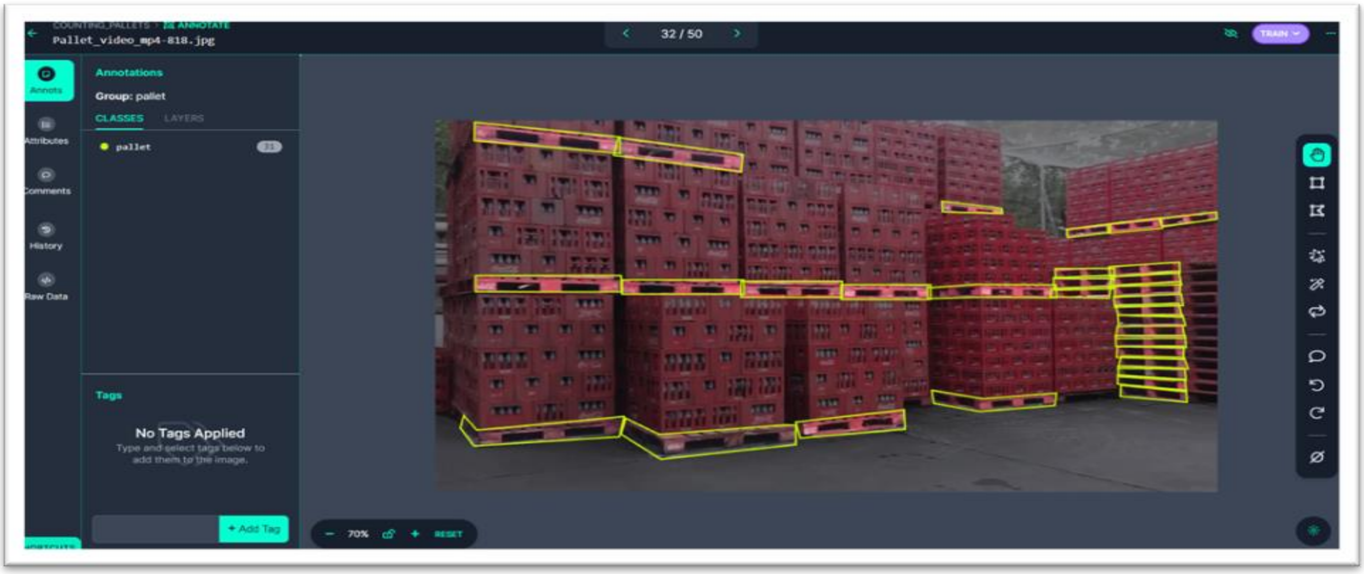

Fig 3 Pallet Annotation in Roboflow: An Overview of the Annotation Process
(Source: Roboflow)

- *Pre-Processing:*

The uploaded image is subjected to basic pre-processing steps and data augmentation:

- ✓ Image resizing: Ensuring uniform dimensions as needed for YOLOv8 to function.
- ✓ Horizontal and Vertical Flip
- ✓ Rotate
- ✓ Shear
- ✓ Data augmentation is done to increase the size of the dataset for training purposes to improve accuracy.
- ✓ Pre-processing
- ✓ Auto-Orient: Applied
- ✓ Resize: Fill (with centre crop) in 640x640

- *Augmentation:*

Augmentation is a transformative technique used to expand a dataset by generating new images from existing ones. By applying various adjustments, slight modifications are made to the original images, enhancing their diversity and usefulness. These augmentation operations collectively create new instances that expose machine learning models to a wider array of scenarios, resulting in improved model robustness and performance when faced with diverse real-world image variations. Augmentation is also done using Roboflow [Fig.4].

- ✓ Flip: Horizontal
- ✓ 90° Rotate: Clockwise, Counter-Clockwise
- ✓ Rotation: Between -15° and +15°
- ✓ Shear: ±15° Horizontal, ±15° Vertical
- ✓ Brightness: Between -5% and +5%
- ✓ Data is then split into Train, Test and Validation sets.

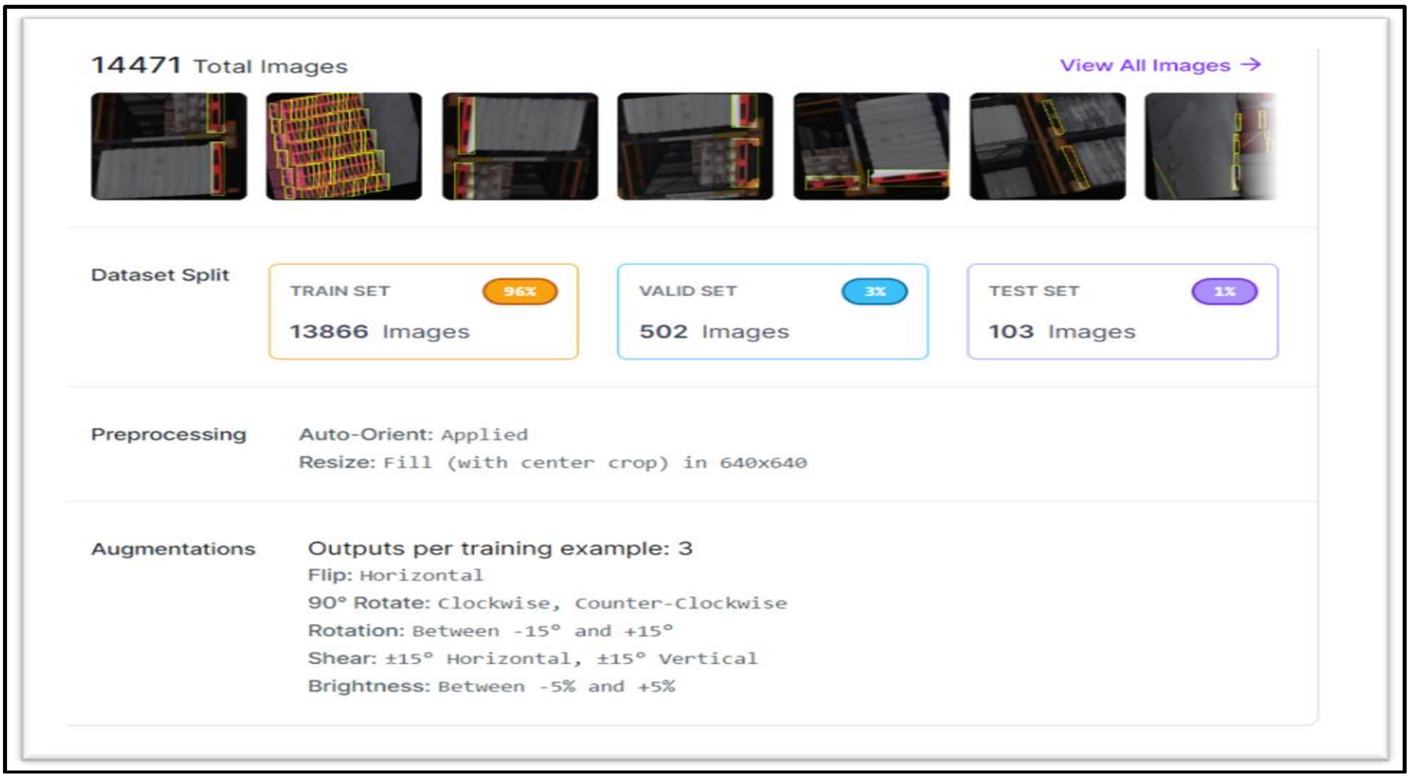

Fig 4 Pre-Processing and Augmentation Steps Performed on Data

➢ *Step 4: YOLOv8 Model Training*

- The YOLOv8 model is then trained using the custom Train dataset.
- After training the model for some epochs and performing validation on the Validation set for fine tuning. [Fig.5]
- The best weights are saved and these weights will be used in further steps.

- *Hyper Parameters used:*

✓ *Data='/content/counting_pallets-3/data.yaml':*

This parameter specifies the path to the YAML file containing configuration details about the training dataset. The YAML file likely includes information such as the paths to training images, annotations, class names, etc. It's a common practice to use YAML files for dataset configuration to keep the code clean and easily configurable.

✓ *Epochs=20:*

This parameter determines the number of epochs for which the model will be trained. An epoch is one complete pass through the entire training dataset. In this case, the model will be trained for 20 epochs.

✓ *Imgsz=640:*

This parameter sets the size to which the input images will be resized before being fed into the model for training. Here, images are resized to a square shape with both width and height set to 640 pixels.

ISSN No:-2456-2165

✓ *Batch=4:*

This parameter specifies the batch size used during training. The training process updates the model's weights based on the average loss calculated over each batch of samples. In this case, the batch size is set to 4, meaning that the model is updated after processing 4 images at a time.

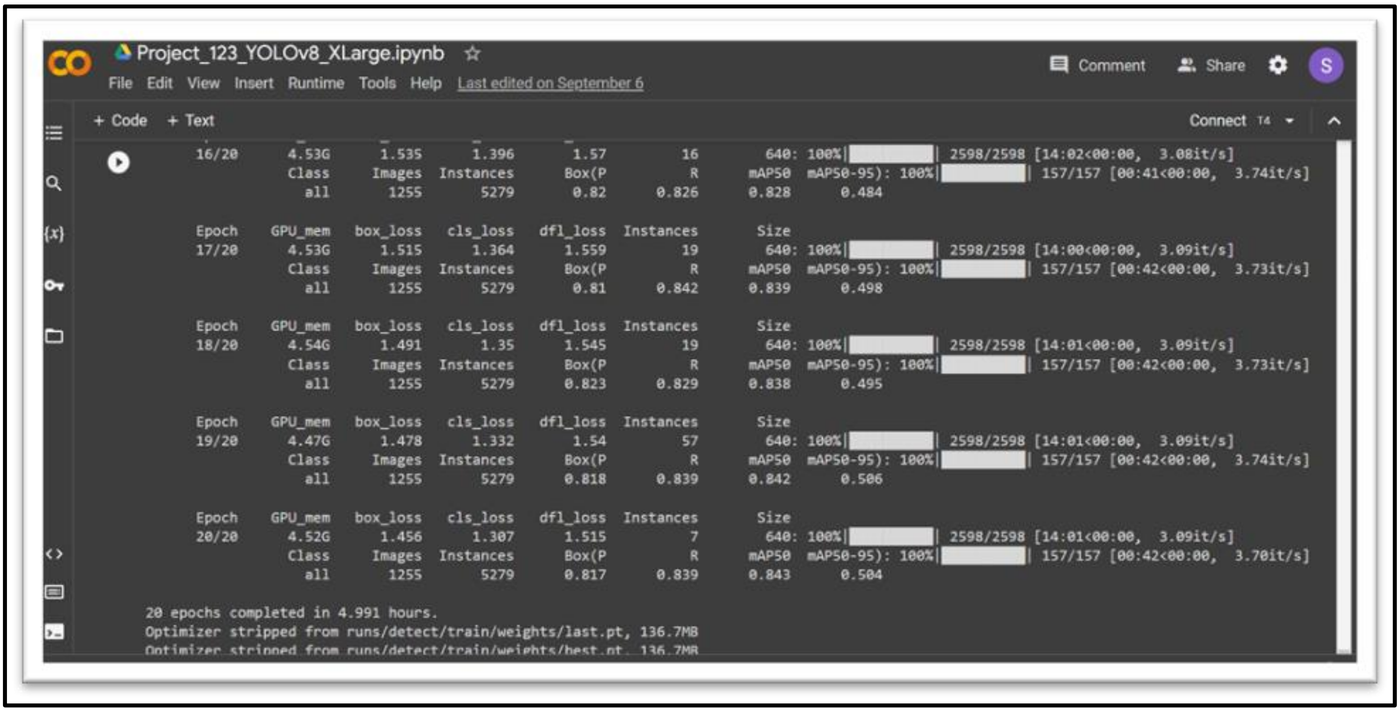

Fig 5 Training of the YOLOv8 Model

➢ *Step 5: YOLOv8 Object Detection using Flask Deployment*

- Deployment is performed using the Flask framework. [Fig.6]
- The YOLOv8 model is now loaded with the best weights from the previous step.
- The model is now given images or videos as input to perform object detection [Fig.7].
- The result is the image or video with bounding boxes drawn over them wherever pallets are detected. The count is given by the number of bounding boxes drawn. [Fig.8]

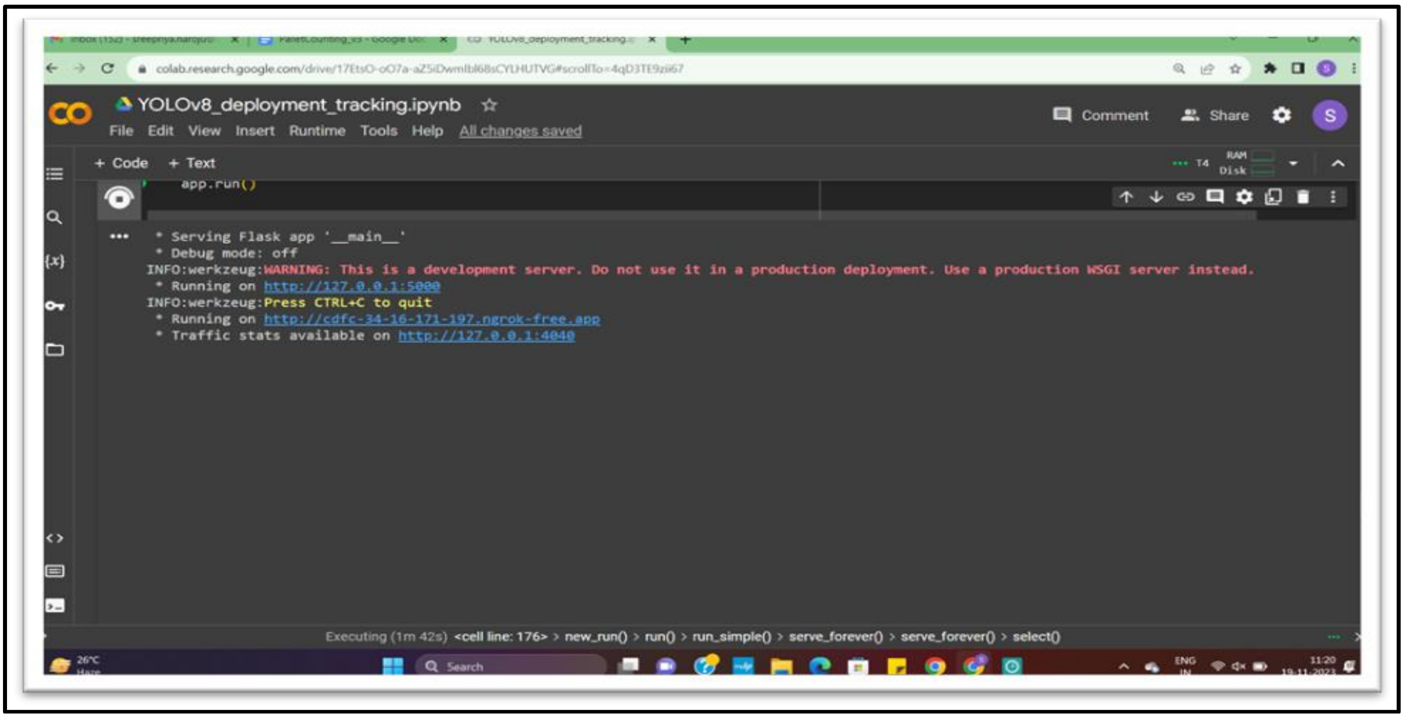

Fig 6 Running Flask Deployment Code

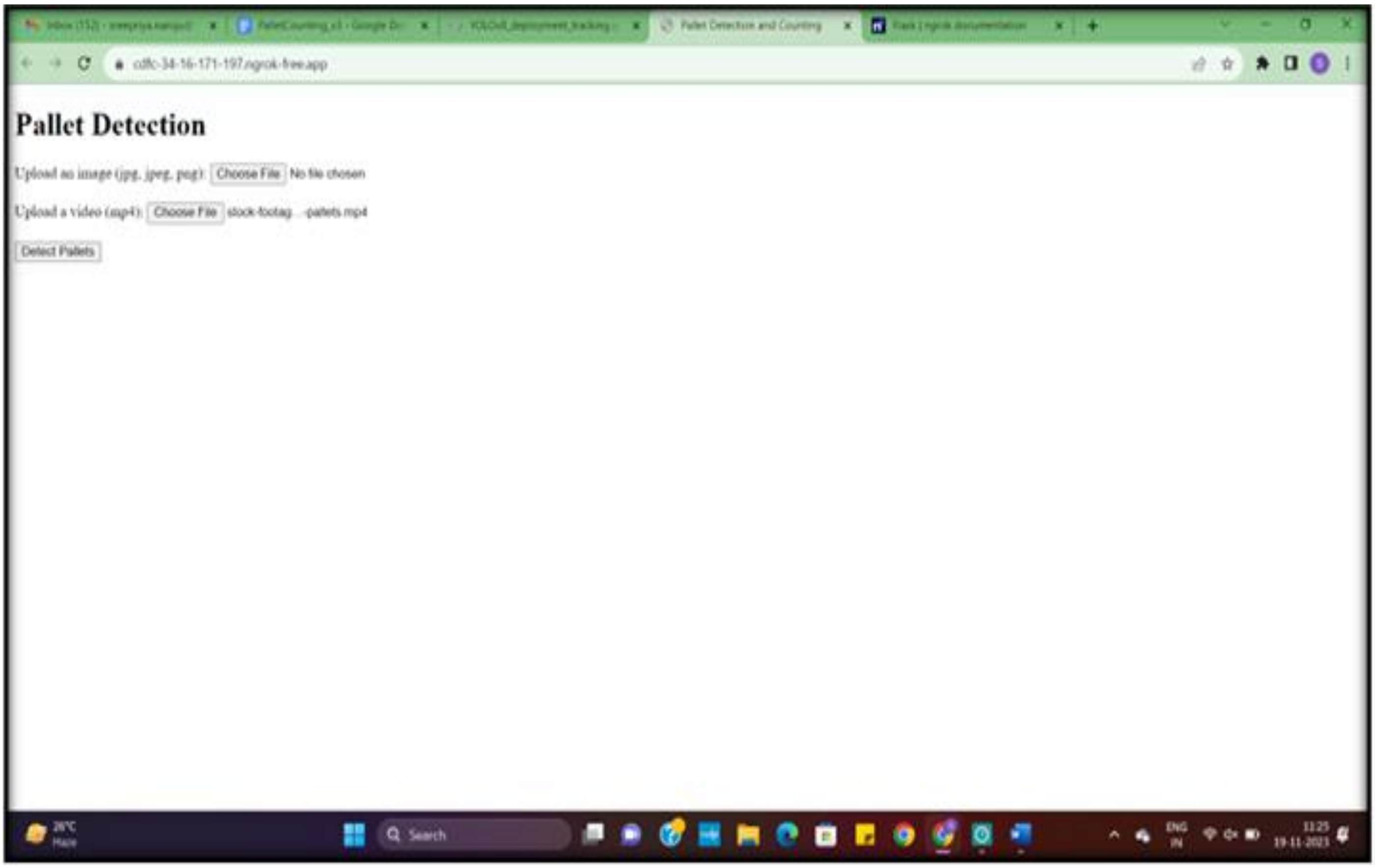

Fig 7 User Interface in Deployment

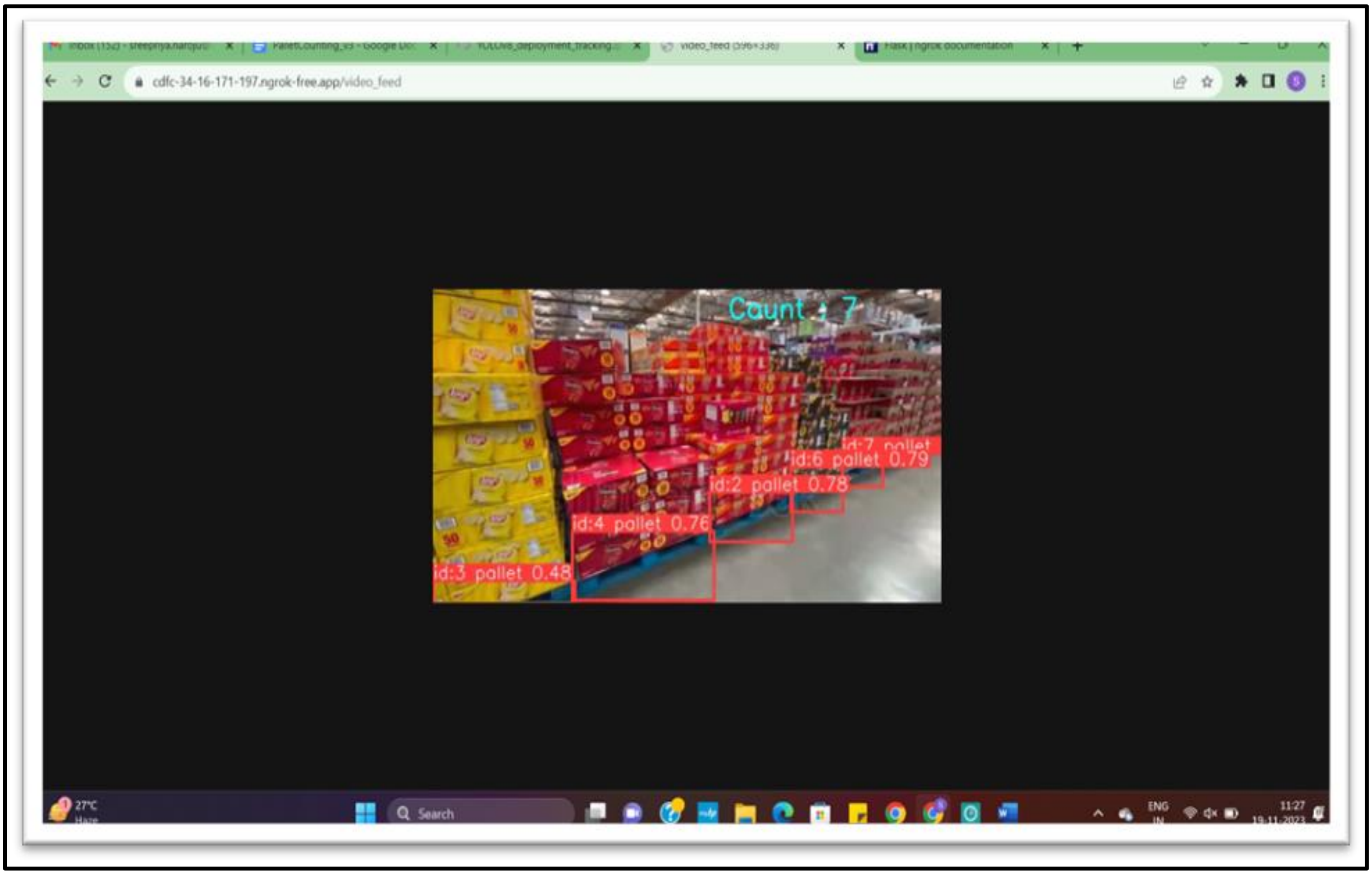

Fig 8 Detecting and Counting the Pallets

III. RESULTS AND DISCUSSION

Object detection systems, such as YOLOv8, often use mAP as a key performance indicator, helping developers and researchers gauge the effectiveness of their models. Mean Average Precision (mAP) is a widely used evaluation metric in the context of object detection and information retrieval. It is especially valuable for assessing the performance of machine learning models that detect and classify objects within images or documents.

In essence, mAP measures the accuracy and precision of object detection models by considering both the precision (correctly predicted objects) and recall (all relevant objects identified) of the model's predictions. It calculates the average precision for each class or category of objects and then takes the mean across all categories, providing an overall assessment of the model's performance.

The application of artificial intelligence and computer vision technologies, specifically the implementation of the YOLOv8 object detection algorithm, yielded promising results in the task of pallet counting in Inventory. The trained model exhibited remarkable speed and precision in accurately detecting and counting pallets. A mean Average Precision (mAP) of 50% was achieved during the training process, indicating a high level of accuracy in object detection.

The object tracking algorithm, similar to byte tracking, effectively maintained the unique identification of detected pallets as those are placed in inventory. This approach significantly improved the precision of pallet counting by avoiding the generation of multiple IDs for the same object, thus minimising counting errors.

In evaluating the object detection models, mAP (mean Average Precision) is a commonly used metric that provides a more comprehensive assessment of model performance. It takes into account both precision and recalls across various object categories. By reporting the achieved mAP of 50%, a more accurate representation of the model's performance in detecting pallets can be provided.

In comparing the achieved mAP value, it is important to note that it can vary depending on the specific dataset and evaluation criteria. However, achieving an mAP of 50% signifies a high level of accuracy and reliability in the pallet detection process.

By employing the YOLOv8 object detection algorithm and implementing object tracking algorithms, the system demonstrated efficient and accurate pallet-counting capabilities in inventory. The achieved mAP of 50% showcased the effectiveness of the model. Future advancements and research in this area hold great potential for further enhancing the accuracy and efficiency of pallet counting systems in inventory management.

IV. CONCLUSION

In conclusion, this research paper demonstrates a concerted effort to address the challenges associated with manual pallet counting in industrial environments. By leveraging the power of deep learning, specifically the YOLOv8 object detection algorithm, the study achieves impressive results in terms of accuracy and efficiency. The integration of advanced technologies, including the Roboflow platform for dataset annotation and Flask for developing a user-friendly web application, enhances the accessibility and practicality of the proposed solution.

The success of the YOLOv8 model in automating pallet counting not only signifies a significant advancement in warehouse management but also highlights the broader transformative potential of deep learning in industrial processes. The study showcases the adaptability and versatility of YOLOv8, emphasizing its applicability beyond pallet counting to various real-world scenarios.

Moreover, the integration of the YOLOv8 model into a user-friendly web application powered by Flask demonstrates a commitment to making advanced technologies accessible and usable for a wider audience. This user-friendly interface contributes to the seamless adoption of automated pallet counting systems, potentially revolutionizing operational standards in the logistics and warehousing sector.

The methodologies employed in this research, such as the CRISP-ML(Q) framework, comprehensive data collection, and the collaborative features of Roboflow, underscore the systematic and thorough approach taken to ensure the success of the project. The study not only delivers a scalable solution for pallet counting but also emphasizes the broader impact of deep learning in optimizing industrial processes.

In essence, this research paper serves as a testament to the transformative power of advanced technologies in the realm of industrial automation. The YOLOv8 model, coupled with innovative tools like Roboflow and Flask, showcases the potential for enhanced efficiency, precision, and cost-effectiveness in various operational domains. As industries move towards a future defined by automation and optimization, this study provides valuable insights into the possibilities and benefits of integrating deep learning techniques into real-world applications.

REFERENCES

[1]. Human errors in warehouse operations: An improvement model, January 2017, International Journal of Logistics Systems and Management 27(3):298,DOI:10.1504/IJLSM.2017.10005117
[2]. Deep learning in computer vision: A critical review of emerging techniques and application scenarios, August 2021, Junyi Chai, Hao Zeng , Anming Li , Eric W.T. Ngai, https://doi.org/10.1016/j.mlwa.2021.100134

[3]. Understanding of Machine Learning with Deep Learning: Architectures, Workflow, Applications and Future Directions by Mohammad Mustafa Taye, Data Science and Artificial Intelligence, Philadelphia University, Amman 19392, Jordan, Computers 2023, 12(5), 91; https://doi.org/10.3390/computers 12050091
[4]. Overview of Image Datasets for Deep Learning Applications in Diagnostics of Power Infrastructure by Bogdan Ruszczak, Paweł Michalski, and Michał Tomaszewski, Department of Computer Science, Opole University of Technology, 45-758 Opole, Poland, Author to whom correspondence should be addressed. Sensors 2023, 23(16), 7171; https://doi.org/10.3390/s23167171
[5]. Revolutionizing Object Annotation For Computer Vision: A Comprehensive Guide to Roboflow, May 2021, Jeng Wei Teoh·
[6]. YOLOX-Ray: An Efficient Attention-Based Single-Staged Object Detector Tailored for Industrial Inspections by António Raimundo, João Pedro Pavia, Pedro Sebastião, and Octavian Postolache, Sensors 2023, 23(10), 4681; https://doi.org/10.3390/ s23104681
[7]. Statistical Analysis of Design Aspects of Various YOLO-Based Deep Learning Models for Object Detection, August 2023, U. Sirisha, S. Phani Praveen, Parvathaneni Naga Srinivasu, Paolo Barsocchi & Akash Kumar Bhoi, https://doi.org/10.1007/s44196-023-00302-w
[8]. A COMPREHENSIVE REVIEW OF YOLO: FROM YOLOV1 AND BEYOND, Juan R. Terven, Diana M. Cordova, October 2023, arXiv:2304.00501v5 [cs.CV]
[9]. Vehicle Detection and Identification Using Computer Vision Technology with the Utilization of the YOLOv8 Deep Learning Method, October 2023, SinkrOn 8(4):2150-2157, DOI:10.33395/sinkron. v8i4.12787
[10]. Efficient Way Of Web Development Using Python And Flask, January 2015, International Journal of Advanced Research in Computer Science, Fankar Armash Aslam, Hawa Nabeel Mohammed, Prashant S Lokhande, Pillai College of Engineering
[11]. Understanding URL Routing in Flask, Kabaki Antony, July 11, 2023
[12]. PYTHON FOR WEB DEVELOPMENT Dr. Uday Patkar (HOD Computer Dept.) Priyanshu Singh; Harshit Panse; Shubham Bhavsar; Chandramani Pandey Department of Computer Engineering, Bharati Vidyapeeth's College of Engineering, Lavale, Pune, DOI: https://doi.org/10.47760/ijcsmc. 2022.v11i04.006
[13]. Flask Python Tutorial for Data Science Professionals, Raghav Agrawal — Updated On June 19th, 2023